OPTICAL CHARACTER RECOGNITION IMPLEMENTATION FOR ADMISSION SYSTEM IN UNIVERSITAS PERTAMINA

Meredita Susanty
Departement of Computer Science
Universitas Pertamina
Email: meredita.susanty@universitaspertamina.ac.id

Herminarto Nugroho
Departement of Electrical Engineering
Universitas Pertamina
Email: herminarto.nugroho@universitaspertamina.ac.id

ABSTRAK

Starting in 2019, prospective college students require to take Computer-Based Writing Exam (UTBK) to register for the state universities in Indonesia. Some private university also adopts this exam as a requirement for admission. One of the private universities that adopt it is Universitas Pertamina. UTBK consist of several exam group score printed in a digital certificate in image format (jpg). The university admission team must download the UTBK certificate that has uploaded by applicants, read and record the score for each exam group then make a calculation to make a decision whether the applicant is accepted in a certain school in the university. It takes 3 to 5 minutes to verify one applicant. Thus, it needs 6.25 days/man to verify one thousand documents. To be able to announce the result within two days, additional personnel is required. As an alternative, this research proposes optical character recognition (OCR) to perform the document verification task. The OCR engine will extract text from an image. Some information from the extracted text is calculated to provide an acceptance decision. The research shows that OCR cannot accurately convert text from an image when there is a grayscale background in the image with an accuracy of 16.67%. However, image preprocessing, which removing the background and non-text object, can improve overall accuracy to 83.33%. Lastly, Tesseract performs better in converting black text with white background than white text with a black background.

Keywords: artificial intelligence; computer vision; pattern recognition; machine learning; optical character recognition

1. INTRODUCTION

Starting in 2019, prospective college students require to take Computer-Based Writing Exam (UTBK) to register for the state universities in Indonesia. UTBK officially replacing the former print-based writing test method.

UTBK consists of Scholastic Potential Tests (TPS) and Academic Competency Tests (TKA) that are in accordance with the exam group of each examinee. The exam group in UTBK is divided into Science and Technology Examination Group (Saintek) with Saintek TKA (Science, Physics, Chemistry, and Biology) and Social and Humanities Examination Group (Soshum) with TKA Soshum exam material (Social Sciences, Geography, History, Sociology, and Economics). Each participant can choose the Saintek and / or Social Sciences exam group. UTBK results will be available ten days after the test. The certificate of result can be downloaded from the website as an image. The certificate then uses as a basis for the admission process in each state university.

ts as a requirement
ina Website. They must fill in the score for each section in the online form and upload the UTBK certificate. The admission team will download the certificate and verify the inputted score from the online form with the score written in the certificate. This verification process usually takes 3 to 5 minutes for each applicant.

Since the admission opens on 14 May 2019 until 29 May 2019, there are 1598 applicants. It is predicted that the applicant will reach around 4000 at the closing of the admission (23 June 2019). The announcement of the admission result within two days after it is closed. Performing verification manually, as described before, will take significant time. It needs 6.25 days/man to verify one thousand documents. In other words, it needs 6 people to complete one thousand document verification in 9 hours.

The manual verification process requires an intensive amount of time and resources. This research proposes an alternative method to verify the document using optical character recognition (OCR). This method has widely used as a form of information entry from printed paper data records and is a common

method of digitizing printed text [1]. OCR engine has been developed into many kinds of domain-specific such as data entry for business documents[2], automatic number plate recognition[3], and converting handwriting[4], [5].

Humans can understand the contents of an image simply by looking. They perceive the text on the image as text and can read it. However, computers do not work in the same way. Sometimes it is difficult to retrieve text from the image because of different sizes, styles, orientation, the complex background of an image. Etc. [6].

This research uses Python-tesseract, an OCR tool for Python[7]. Tesseract [8] is one of the most accurate open source OCR Engines [9]. It is written in C++. Python-tesseract is a wrapper for the Tesseract OCR engine. The success of applying OCR in converting data from UTBK certificates in image format will significantly reduce the time to perform verification and improving the efficiency of workflow.

2. RESEARCH METHODOLOGY

This research follows a sequence of activities, as shown in Figure 1, which consists of collecting data for validation, developing a prototype for converting the UTBK certificate, and testing the prototype.

Figure 1. Research Methodology

Data collection intended to collect a small set of data to test and evaluate the performance of the OCR engine that is used. During development, it is important to understand how to access tesseract OCR via the python programming language and define the algorithm to perform certificate verification.

Once the prototype is developed, it is tested with various cases to identify the limitation of this method. This limitation will then be used to restrict the acceptable certificate uploaded.

3. RESULT AND DISCUSSION

Data for validation is collected from the Internet. Because the prototype will be used to read the UTBK certificate that has a standard format, this research only uses one image as a sample, as shown in Figure 2. The image is rotated 90-degree, and 180-degree then save as a new file. The validation testing shows that the Tesseract engine able to identify objects even though it is upside down or 90-degree rotated. However, the result does not construct a word, as shown in Figure 3(b) and Figure 3(c).

Figure 2. Sample Data From The Internet [10]

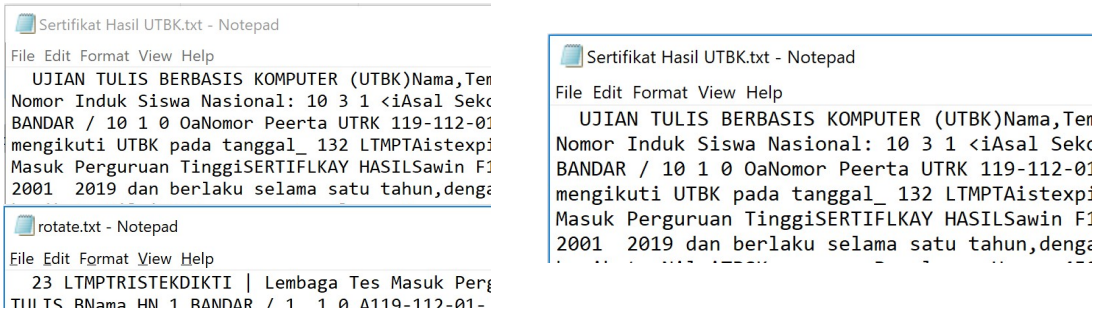

Figure. 3(a). Original Image to Text Conversion Result (B) Rotated 90-Degree Image to Text Conversion Result. (C) Rotated 180-Degree Image to Text Conversion Result

There is a potential where the applicant uploads the certificate from various angles. Hence, as shown in Figure 4, the algorithm for the prototype check the dimension of the uploaded image. The standard orientation of the UTBK certificate is a landscape, which means the image is wider than its height. When the uploaded image is taller than its wide, it is then rotated 90-degree. Although its is rotated 90-degree, there is still a possibility that the image is upside down. Upside down image will be identified after the image to text conversion result is compared with a list of common words (sertifikat, hasil, ujian, tulis, berbasis, komputer, nama, tempat, tanggal, lahir, nomor, etc.) from the UTBK certificate. When there are no common words in the resulted text, the uploaded image is rotated 180-degree. The rotated image then reconverted. The last step is identifying the following data; name, place, and date of birth, national student number, high school and its identification number, exam number, and score for each exam section, and loads it into a csv file.

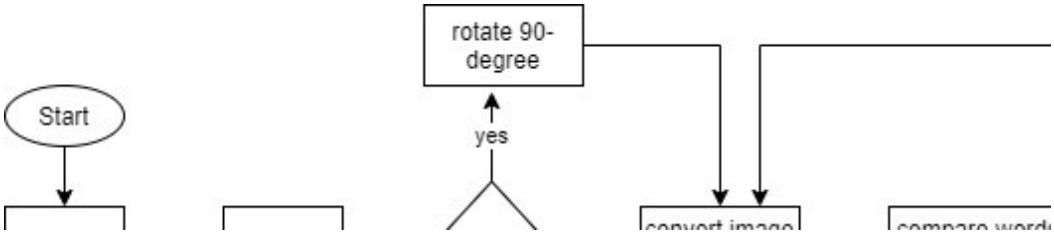

Figure. 4. The Algorithm For The Prototype

This research uses 129 UTBK certificates uploaded by Universitas Pertamina's applicant for testing the prototype. The size of the files is ranging from 20 KB to 199 KB. The dimension of the image also ranging. Only the image with 2339 width and 1654 height that can be converted correctly. However, the images with the same dimension, as shown in Figure 5, might have a different order of the words, as shown in Figure 6.

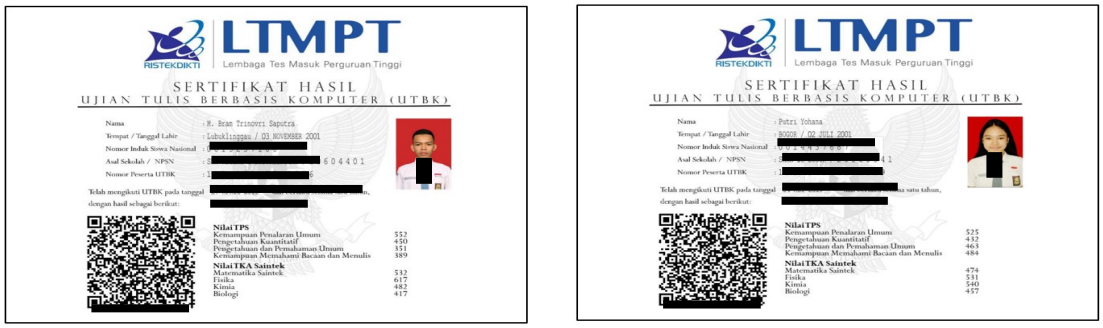

Figure. 5. Source Images

Tesseract was originally designed to recognize English text only. Tesseract can only handle left-to-right language [11]. However, the conversion result in Figure 6 (left) shows that it converts text from top to bottom in one block then continues to the next block from top to bottom.

Figure. 6. Converted Results

This research performs preprocessing to improve the accuracy of the Tesseract engine. The aim of preprocessing is to take out as much noise as possible. Referring to Figure 5, the grayscale background, photo, and QR code are considered as noise. Removing noises from an image involves a lot of trial and error. Figure 7 shows the preprocessing algorithm. Next to some steps in the algorithm is a sample of the image that uses as an input for the algorithm. The converted text result for some images shown at the right of each image.

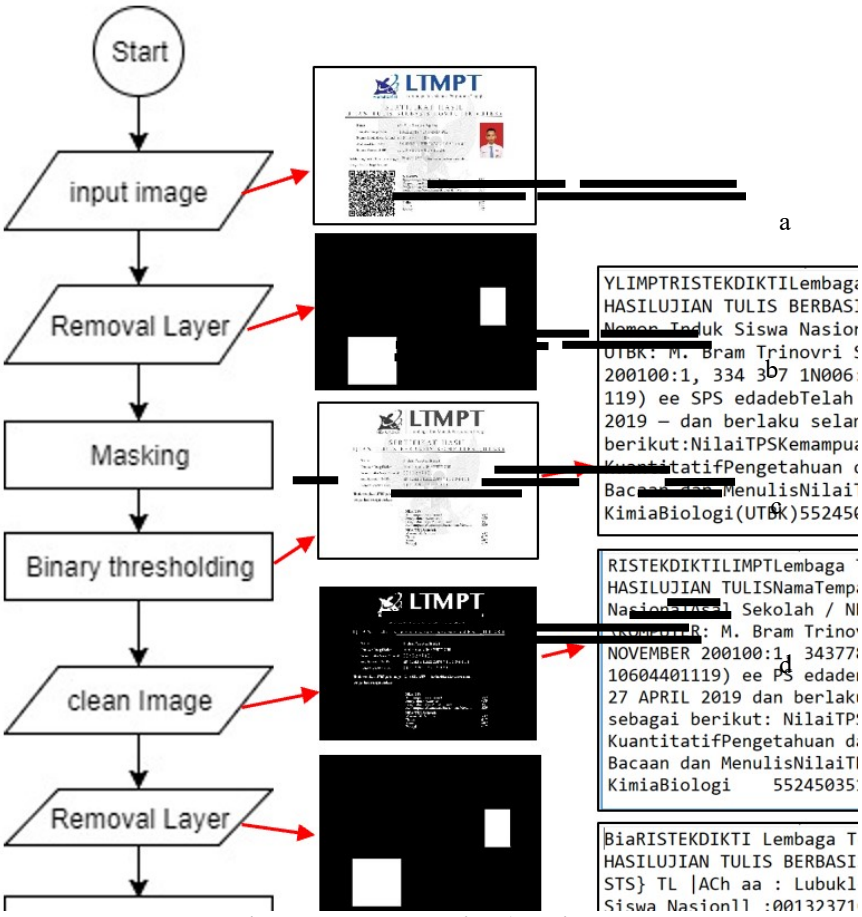

Figure. 7. Pre-Processing Algorithm

The preprocessing algorithm, as shown in Figure 7, starts with removing photo and QR code by adding the original image with a black image with a white block in the QR code and photo area. Adding two images takes two identically sized images and produces a third image of the same size as the first two as an output, in which each

pixel value is the sum of the values of the corresponding pixel from each of the two input images. The pixel value for black is 0, and white is 255. Adding the original image with a black background resulting in the original color because adding pixel color with 0 does not change the color. On the other hand, adding an original image with white removes the original color. It is because the maximum pixel value is 255. The text conversion result from an image without photo and QR code still shows an incorrect order.

Next, the preprocessing step aims to remove the background. Thresholding is the simplest method to separate regions, which is higher than the set threshold[12], [13]. If pixel value is greater than a threshold value, it is assigned one value, otherwise it is assigned another value. Because this step intends to remove grayscale background, the threshold is 185, the gray pixel is changed to black, while black pixel is changed to white. However, the text conversion result from image without photo and QR code still show an incorrect order.

The next trial is to re-run the preprocessing step so that it provides an image similar to the original image in back and white, without background, photo, and QR code. Photo and QR code are replaced with a black box. The conversion from image to text shows an improved accuracy compared to other preprocessing steps result. All conversions are compared with each other to calculate the accuracy of the converted text, as shown in table 2. The conversion result from the unprocessed image has 21 new words and 54 missing words compared to the correct result (training). The total difference (addition + remove) divided by the total word shows the accuracy of the conversion, as shown in table 2.

Table 1. Converted text comparison

	unprocessed	***Figure 7.a***	***Figure 7.b***	***Figure 7.c***	***Figure 7.d***	***Training***
unprocessed		*17*	*50*	*52*	*50*	*54*
Figure 7.a	*9*		*23*	*23*	*22*	*20*
Figure 7.b	*21*	*44*		*8*	*1*	*5*
Figure 7.c	*23*	*43*	*8*		*8*	*10*
Figure 7.d	*21*	*46*	*4*	*11*		*5*
Training	*21*	*45*	*10*	*14*	*0*	

Description:
White background: Number of words added
Grey background: Number of words removed/missing

Table 2. Converted text accuracy

Converted Text	***Accuracy***
Training	*100%*
unprocessed	*16.67%*
Figure 7.a	*27.78%*
Figure 7.b	*83.33%*
Figure 7.c	*73.33%*
Figure 7.d	*83.33%*

Based on the accuracy test, it is shown that figure without a background image (Figure.7.b and Figure.7.d) has the highest accuracy. This research also shows that Tesseract has better accuracy in converting black text with white background than white text with black background. It is supported by the fact that although Figure.7.b and Figure.7.d has the similar accuracy the order of the words is different. Furthermore, some objects in the image (photo and QR code) have an impact on the text conversion order. However, this research cannot find the relation. This issue is left for further research.

4. CONCLUSION & FUTURE WORK

OCR engine can help to automate the process of reading a digital certificate. Although it is stated that the OCR engine read the text in the digital image from left to right, in some cases, it read text from top to bottom. Preprocessing, intended to remove any noises from the image, improves the accuracy of the OCR engine. However, there are still some limitations; the correct conversion only applied for the image with 2339 x 1654 dimension, photo and QR code must be replaced with a black box to maintain a consistent order of text conversion. Considering these limitations, the admission website should warn each applicant about the accepted image size and check the dimension of the uploaded image. Further research intended to explore the correlation between colour in the non-text object (photo and QR code) with the order of text conversion.

REFERENCES

[1] S. Mori, H. Nishida, and H. Yamada, *Optical character recognition*. J. Wiley, 1999.
[2] J. M. White and G. D. Rohrer, "Image Thresholding for Optical Character Recognition and Other Applications Requiring Character Image Extraction," *IBM J. Res. Dev.*, vol. 27, no. 4, pp. 400–411, Jul. 1983.
[3] M. T. Qadri and M. Asif, "Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition," in *2009 International Conference on Education Technology and Computer*, 2009, pp. 335–338.
[4] N. Arica and F. T. Yarman-Vural, "Optical character recognition for cursive handwriting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 801–813, Jun. 2002.
[5] A. N. Bhute and B. B. Meshram, "Text Based Approach For Indexing And Retrieval Of Image And Video : A Review," *Adv. Vis. Comput. An Int. J.*, vol. 1, no. 1, pp. 27–38, 2014.
[6] C. I. Patel, A. Patel, and D. Patel, "Optical Character Recognition by Open source OCR Tool Tesseract : A Case," vol. 55, no. 10, 2012.
[7] "pytesseract 0.2.6," 2018. [Online]. Available: https://pypi.org/project/pytesseract/. [Accessed: 30-May-2019].
[8] "Tesseract OCR – opensource.google.com." [Online]. Available: https://opensource.google.com/projects/tesseract. [Accessed: 30-May-2019].
[9] "OCR - Community Help Wiki." [Online]. Available: https://help.ubuntu.com/community/OCR. [Accessed: 29-May-2019].
[10] "Sertifikat Nilai Hasil UTBK 2019/2020 | Soal UTBK SBMPTN 2019 dan Pembahasan [Prediksi]." [Online]. Available: https://www.e-sbmptn.com/2019/04/sertifikat-nilai-hasil-utbk.html. [Accessed: 30-May-2019].
[11] Shreeshrii, "Tesseract - Background and Limitations," 2018. [Online]. Available: https://github.com/tesseract-ocr/tesseract/wiki/TrainingTesseract2#background-and-limitations. [Accessed: 30-May-2019].
[12] C. Cahyono, G. Prasetyo, A. Yoza, and R. Hani, "Multithresholding In Grayscale Image Using Pea Finding Approach And Hierarchical Cluster Analysis," *J. Ilmu Komput. dan Inf.*, vol. 7, no. 2, p. 83, Aug. 2014.
[13] R. C. Gonzalez and R. E. (Richard E. Woods, *Digital image processing*. Prentice Hall, 2008.